

PROYECTO DE SISTEMAS INFORMÁTICOS

FACULTAD DE INFORMÁTICA

UNIVERSIDAD COMPLUTENSE DE MADRID



CURSO 2010/2011

**Instrumento de planificación para la
toma de imágenes satelitales
a través de Matlab y Google Earth**

PROFESOR DIRECTOR: Dr. Segundo Esteban San Román

AUTOR: Antonio Morelli

El abajo firmante autoriza la Universidad Complutense a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor, tanto la propia memoria, como el código, la documentación y/o el prototipo desarrollado.

Fdo. Antonio Morelli

Resumen:

Este proyecto desarrolla una representación 3D en alto grado de detalle de la planificación de toma de imágenes por satélite. El proyecto está desarrollado sobre Matlab® y utiliza como herramienta de representación 3D Google Earth® para la representación en alto grado de detalle. Este proyecto da un acabado gráfico a un optimizador de la planificación orbital, que es otro proyecto de Ingeniería Electrónica de la Universidad Complutense de Madrid.

Palabras clave:

Informática Gráfica, Toma de Imágenes por Satélite, Matlab, Google Earth, Dinámica Orbital.

Abstract:

This Project develops a high detail 3D representation of satellite image capture planning. The project is developed using Matlab ® as main API. Google Earth ® API is used for a 3D representation with high detail degree. This project is connected with an orbital planning optimizer, developed as other project of the Electronic Engineering of the Complutense University of Madrid.

Keywords:

Computer Graphics, Satellite Image Capture, Matlab, Google Earth, Orbital Dynamic.

Índice

1. Presentación y Objetivos del Proyecto	9
1.1 Introducción	9
1.2 Ojetivos del proyecto.....	11
1.3 Instrumentos usados	11
1.3.1 Matlab	12
1.3.2 Google Earth.....	13
1.4 Planificación del Proyecto	14
2. Desarrollo del Proyecto.....	17
2.1 Documentación y aprendizaje.....	17
2.2 Fusión de Matlab y Google Earth	20
2.3 Proyecto	21
3. Conclusiones.....	29
3.1 Futuras expansiones.....	33
4. Bibliografía	35

1. Presentación y Objetivos del Proyecto

1.1 Introducción

Con el término “satélite artificial” se hace referencia a una masa de materia en órbita alrededor de un cuerpo celeste, puesta por el hombre a través del uso de medios tecnológicos y con el fin de apoyar ciertas necesidades humanas.

El éxito de los satélites está en el hecho de que no necesitan casi mantenimiento y a menudo consiguen alimentarse autónomamente a través de la energía solar.

A través de ellos podemos observar nuestro planeta desde una altura que nos permite sacar datos e información en muy poco tiempo. Además nos permiten comunicarnos muy rápidamente con todo el planeta, sin costes de instalación de cables y repetidores en tierra.

Los primeros satélites artificiales se enviaron al espacio durante la segunda guerra mundial (1939-1945). En los años siguientes, los EEUU y la Unión Soviética lucharon por décadas para el control de la exploración espacial. En un principio lo lograron los rusos y en 1957 enviaron en órbita el satélite artificial Sputnik I, que permaneció en órbita durante 92 días; y el mismo año enviaron en órbita el primer ser viviente: Lajka, una perrita.

Uno de los usos principales de los satélites artificiales es en el campo de las telecomunicaciones, sin embargo existen varios tipos de satélites con objetivos distintos: tele-relevación, navegación, militares y meteorológicos.

El primer satélite meteorológico, TIROS1 (Television Infrared Observation Satellite), fue lanzado por la NASA en 1960 y proporcionaba información sobre las condiciones meteorológicas mundiales.

Después de este test, la NASA siguió desarrollando nuevos tipos de satélites cada vez más avanzados, estipulando acuerdos con las mayores potencias europeas.

En 1969 EEUU estableció su primacía en la historia de la industria, enviando en la órbita “Apollo 11”, que aterrizó en la luna dándole la oportunidad a su tripulación de mover los primeros pasos en la superficie lunar.

Desde los años setenta hasta el día de hoy ha habido un fuerte aumento de los envíos de satélites artificiales y EEUU y Europa tiene la indiscutible primacía. El estudio del movimiento de estos satélites es objeto de estudio por la astrodinámica.

El movimiento y el trayecto de un cuerpo en el espacio, inclusive de los satélites, se llama órbita. Un satélite que gira alrededor de la tierra es sujeto a dos tipos de fuerzas: la fuerza de atracción de la gravedad que los empuja hacia la tierra y la fuerza centrífuga que en cambio lo rechaza.

Mi proyecto nace a partir de los resultados de un trabajo anterior en el que se estudiaba el problema de “la planificación orbital del satélite”, es decir calcular evolución orbital del satélite y planificar el modo de operación. Todo parte de un TLE(Two Element Sets), que está compuesto por dos líneas de códigos que representan los valores matemáticos que definen la órbita del satélite y está contenido en un documento en formato ASCII.

Una vez obtenida las coordenadas de la órbita del satélite, el proyecto dibujaba a través de un simple *plot* sobre Matlab la órbita del satélite y una zona roja con el objetivo de indicar el área en el que se efectuaba la toma de las imágenes por parte del satélite.

El proyecto tomaba en cuenta también el problema del tiempo. En particular se aplicaba para un satélite con un periodo orbital de 90 minutos, superponiéndose varias órbitas, para realizar el cálculo más realístico.

Este proyecto se desarrolla a como una aplicación que permita la visualización 3D de la planificación del satélite, partiendo del cálculo del TLE, y de una zona de toma de imágenes, donde el satélite sacará una series de fotos.

La diferencia y el carácter innovador de este proyecto está en el uso, para la representación del movimiento del satélite y para todo lo dicho anteriormente, de la aplicación de Google Earth.

1.2 Ojetivos del proyecto

A partir del anterior proyecto, que consistía en el cálculo y en la representación del trayecto satelital, se ha tomado la decisión de representar todo a través de la unión de dos aplicaciones: Matlab e Google Earth.

Gráficamente los objetivos son varios:

- Representación de la trayectoria del satélite (Path).
- Representación de bases de comunicación en tierra
- Representación de una zona de toma de imágenes
- Representación de un área dentro de la zona anterior, que señale el espacio exacto de la toma de imágenes por parte del satélite.
- Coloración de forma distinta de la traza en tierra.
- Generar un 'Tour', una simulación real del movimiento del satélite, poniendo particular cuidado en el tiempo, para que el movimiento del satélite en su propia órbita sea lo más realísticos posible.

1.3 Instrumentos usados

En el desarrollo del proyecto se ha utilizado:

- ◆ Matlab ®
- ◆ Google Earth ®

El primero es una herramienta-lenguaje de soporte para el cálculo científico que tiene que ver con el desarrollo, la implementación y el análisis de los algoritmos numéricos para el uso de modelos matemáticos.

El segundo es una API de gráficos 3D que ofrece la posibilidad de “ponerse alas”, de atravesar el océano en pocos segundos, de mirar el planeta desde Km de altura o hacer una ampliación hasta la superficie de la tierra, para tener una información lo más detallada posible.

1.3.1 Matlab

En 1980 Cleve Moler escribió una primera versión de Matlab con el objetivo de desarrollar una calculadora matricial interactiva. De hecho, Matlab del que deriva el acrónimo Matrix Laboratory, era una sofisticada calculadora científica que operaba sobre matrices. En 1985 fundó la sociedad “The Mathworks” y Matlab se convirtió en un lenguaje orientado a las matrices.

Matlab es una lenguaje con un nivel alto y un ambiente interactivo capaz de hacer cálculos más rápidamente con respecto a los lenguajes de programación tradicionales como C, C++ e Fortran.

Tiene además complejos sistemas de simulación y para la creación de modelos como Simulink.

Matlab se utiliza sobre todo para el cálculo numérico y para el análisis estadístico, que incluye el lenguaje de programación creado por Mathworks. Permite visualizar funciones y datos, operar con matrices, implementar algoritmos, crear una interfaz entre usuarios y con otros programas.

La sucesión de órdenes usados para solucionar algoritmos y problemas, pueden ser memorizados en un documento de texto. De esa manera se crear un M-file Matlab. Existen dos tipos de M-file:

- Script M-file: permite realizar una secuencia de o instrucciones Matlab. Suele usarse para que la repetición de una secuencia de operaciones sea automática o cuando se quiere realizar un proyecto que supone cierta complejidad.

- *Function* M-file: acepta al entrar algunos datos y produce otros a la salida. Las variables de una *función* son, por defecto locales, para compartirlas con el ambiente principal de Matlab o con otras *funciones*, es necesario especificarla de forma explícita como globales.

Matlab tiene una característica importante: es una plataforma múltiple, es decir que funciona con varios sistemas operativos como Windows, Linux, Mac OS e Unix.

1.3.2 Google Earth

El programa con el cual he decidido enlazar Matlab, es Google Earth.

Google Earth es un software que genera representaciones virtuales de la Tierra utilizando imágenes satelitales, fotografías aéreas y datos topográficos memorizados en una plataforma GIS (Geographic Information System).

Una plataforma GIS es un sistema informático que permite la adquisición, la registro, la análisis, la visualización y la restitución de información que deriva de los datos geográficos.

El programa en principio se llamaba Earth Viewer, sucesivamente en el 2004 la sociedad que lo creó, la KeyHole, fue adquirida por Google. A través de una fusión entre Earth Viewer y Google Maps, gracias a la cual se podían utilizar las mapas y toda la información de los servicios Maps, el programa fue renombrado y ahora está distribuido gratuitamente por la Sociedad.

También Google Earth es una multiplataforma; esto quiere decir que requiere un sistema operativo como Windows, Mac OS o Linux.

Se trata de una aplicación gráfica tridimensional que permite desarrollar diferentes operaciones: es posible visualizar determinadas áreas del planeta con una cierta resolución gráfica, que varía según la importancia del lugar (en las ciudades más grandes de hecho es posibles ver edificios,

carreteras y coches); se puede explorar el mundo como nunca antes, viajar atrás en el tiempo gracias a las imágenes históricas, sumergirse en las profundidades del océano, mirar las estrellas, las galaxias y las constelaciones o registrar una ruta para poderlo volver a ver y compartir sucesivamente.

Al abrir el programa, será visualizada la tierra en su totalidad; sucesivamente es posible realizar una búsqueda, insertando las coordenadas geográficas o la dirección de lo que estamos buscando, o hacer girar la tierra utilizando el ratón y determinadas combinaciones de los dígitos para llegar al lugar deseado.

Los mapas visualizados están proporcionadas por entidades importantes como la NASA, o la Digital Globe. Gracias a la información hecha pública, podemos tener, en las ciudades, resoluciones más importantes que las de pixel por metro.

Los lugares con una más alta resolución son los Estados Unidos y Europa, aunque la actualización de los datos mejora año tras año.

Los problemas relacionados con una escasa resolución, pueden surgir porque se utilizan múltiples fuentes o por inexactitudes debidas a cálculos imprecisos, incluso por errores humanos. Se añaden dificultades debidas a la unión de fotos que a lo mejor han sido sacadas en años diferentes o problemas relacionados con la seguridad internacional; de hecho Google, en algunos casos, ha tenido que censurar fotos de localidades que no se han podido publicar.

1.4 Planificación del Proyecto

He abordado el estudio de este proyecto gracias a la documentación, al análisis y test sobre las dos herramientas utilizadas. El trabajo se ha desarrollado en 3 etapas fundamentales:

1. Documentación y Aprendizaje
2. Fusión de Matlab y Google Earth
3. Proyecto

2. Desarrollo del Proyecto

2.1 Documentación y aprendizaje

El primer paso para abordar el estudio de este trabajo, ha sido analizar el proyecto por el cual tendría que haber empezado para luego convertirlo en lo que es ahora.

“Software de tierra para la planificación de la actitud del satélite orbital”; este es el título del proyecto al cual he hecho referencia.

Se aplicado a un satélite LEO (Low Earth Orbit) o sea un satélite que tiene una órbita terrestre baja. Es una órbita circular entre la atmósfera y los cinturones de Van Allen, a una altura que varía entre los 200 y los 2000 km. Este cuerpo que orbita LEO tiene un tiempo de revolución de alrededor de 90 minutos con una velocidad de 27.400Km/h más o menos.

El primer objeto del estudio es el cálculo de la órbita.

Para especificar las coordenadas inerciales ECI (Earth Centered Inertial) de un satélite en su órbita, a menudo se utilizan los elementos Keplerianos. Se trata de conjuntos de seis parámetros que proporcionan una descripción completa y geométrica de la órbita. Estos elementos son indicados en la Figura 1:

- excentricidad (e)
- semieje mayor (a)
- ascensión del nodo ascendente (Ω)
- inclinación del plano orbital (i)
- argumento del perigeo (ω)
- tiempo de pasaje a través perigeo (t_p) .

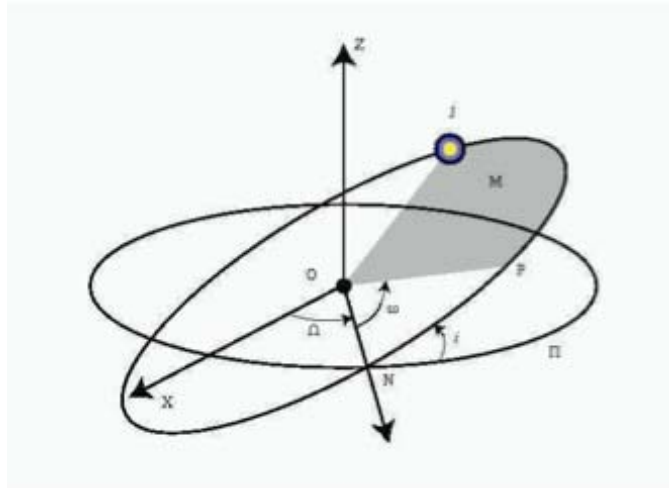


Figura 1-Elementos Keplerianos

Los parámetros a y e proporcionan la forma exacta y el tamaño de la elipse que constituye la órbita; los otros cuatro parámetros proporcionan la información sobre la orientación de la órbita en el espacio, indicado por Ω y i , y sobre la posición del satélite, que se obtiene con ω e tp .

Una vez que estos seis parámetros estén claros, se puede obtener la órbita del satélite, o sea la evolución de la posición del satélite en el tiempo.

Un aspecto importante es que los elementos Keplerianos son unas medias obtenidas eliminando variaciones periódicas en las cantidades físicas a las cuales se refieren. Por esta razón no todos los métodos de propagación tienen las condiciones para construir las variaciones periódicas.

El modelo de propagación que utilizaremos es el modelo.

Juntos con los seis elementos Keplerianos, el propagador es utilizado para describir la posición del satélite en su órbita en un determinado momento y está generalmente codificado para obtener una descripción lo más completa posible del satélite al cual se refiere. Uno de los estándares más comunes para su descripción es el formato TLE (Two-Line Element) que es específico para NORAD y es utilizado por NORAD y por la NASA.

El formato TLE se compone de dos líneas de sesenta y nueve caracteres cada una, en las cuales están codificados los elementos Keplerianos que definen la órbita. Los caracteres utilizados normalmente son los números de cero a nueve, las letras mayúsculas, el espacio, el punto, los signos y los signos más y menos.

NORAD mantiene un registro que actualiza periódicamente con el TLE de aquellos satélites que orbitan alrededor de la Tierra. Esta información, excepto la de los satélites que por motivos estratégicos o militares tiene que mantenerse secreta, se proporcionan de forma gratuita.

La información contenida en el TLE de los satélites en órbita alrededor de la tierra, se puede encontrar en el sitio: <http://celestrak.com/NORAD/elements/>.

El siguiente paso, después de haber recibido y decodificado el TLE que define la órbita del satélite, es su propagación para obtener la posición del mismo en cada instante del tiempo.

Una de las cuestiones más importantes en la caracterización de las posiciones de los satélites, es aquella relacionada con los sistemas de referencia utilizados y las transformaciones que deben ser realizadas de un sistema a otro.

En realidad estamos trabajando sobre un sistema tridimensional que puede asumir diferentes sistemas de referencia, primero porque obligados, y también por conveniencia. Es entonces importante saber realizar las transformaciones adecuadas para expresar la misma grandeza en los diferentes sistemas.

A lo largo del desarrollo del proyecto, el sistema de referencia utilizado será aquello ECI, caracterizado por los siguientes puntos:

- Origen de la Tierra
- Eje OZ que apunta hacia el norte a lo largo del eje de rotación de la tierra
- Eje OX en la dirección del equinoccio de primavera
- Eje OY que forma un sistema dextrorso (horario).

Para el proyecto necesitamos entonces obtener una órbita especificada por el TLE. Aquella elegida es una órbita a partir del siguiente TLE:

1	88888U		80275.98708465	.00073094	13844-3	66816-4	0	8
2	88888	72.8435	115.9689	0086731	52.6988	110.5714	16.05824518	105

Para tener una idea de como será la órbita, dibujamos sobre un mapa la proyección del camino del satélite. El resultado puede ser visualizado en la Figura 2:

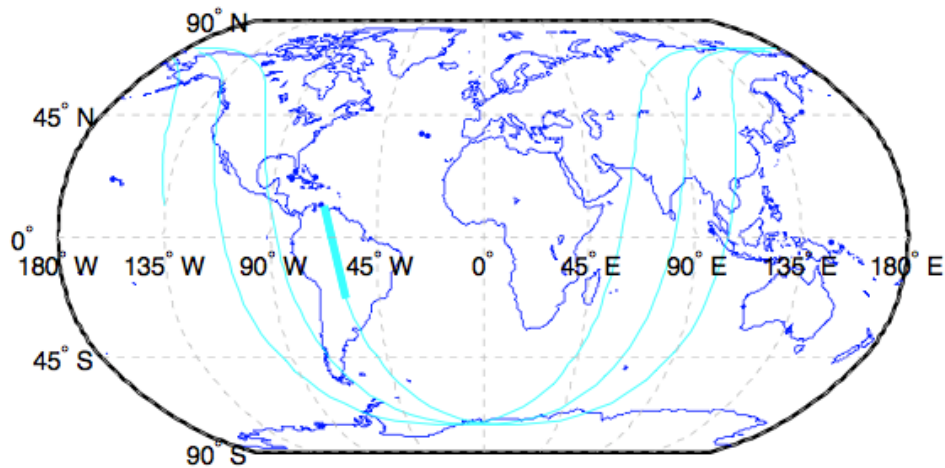


Figura 2- órbita elegida

2.2 Fusión de Matlab y Google Earth

El segundo punto importante en el desarrollo de mi tesis es la fusión y el estudio de las dos herramientas utilizadas: Matlab y Google Earth.

Después de haber estudiado por separado los dos y haber profundizado, evaluado, buscado, analizado y aprendido su funcionamiento, he conseguido fusionarlos.

El primer paso para la fusión, ha sido la realización, a través de Matlab, de un fichero *kml* .

KML (Keyhole Markup Language) es un lenguaje de markup XML para la anotación sobre mapas de marcadores, formas poligonales, recorridos, modelos 3D, descripciones y etiquetas. En principio ha sido desarrollado para utilizarlo con Google Earth pero, desde entonces, ha sido adoptado y utilizado por muchas aplicaciones relacionadas con un mapa, por ejemplo Google maps o Google Mobile. Keyhole es un nombre antiguo que proviene del software del cual procede Google Earth.

En el fichero principal *Antonio*, están indicadas las diferentes funciones y llamadas a otras funciones:

```
full_path = demo_ge_metodo('kml');
```

entonces se abre automaticamente el fichero kml a través de esta línea:

```
ge.OpenKmlFile(full_path,true);
```

esta operación se repite con todas las funciones presentes y necesarias al desarrollo de la aplicación.

Las funciones mencionadas son del tipo: *demo_ge_metodo*; éstas recuerdan a la vez dos métodos del tipo: *ge_metodo* y *ge_output*.

El primero es el verdadero metodo, lo que se encarga de crear y escribir el cuerpo del fichero kml.

El segundo es un método utilizado de forma igual para todas las funciones, y se encarga de escribir la cabecera y el pie de un fichero kml.

2.3 Proyecto

Analizamos ahora el Proyecto, sus funciones, los diferentes métodos y los resultados gráficos obtenidos. Siguiendo un orden cronológico, recorreremos el camino empleado por la creación de la aplicación.

Demo_ge_percorso: es el método para la creación del recorrido, el así llamado Path. En la representación será visualizado el recorrido en tierra, para poder mejor comprender a lo largo de cual de ellos se moverá el satélite.

Este método permite también la marcación con color del recorrido, según las diferentes zonas: captación de las imágenes (amarillo), estaciones en tierra (negro), recorrido normal (azul).

Podemos ver el resultado en la Figura 3:

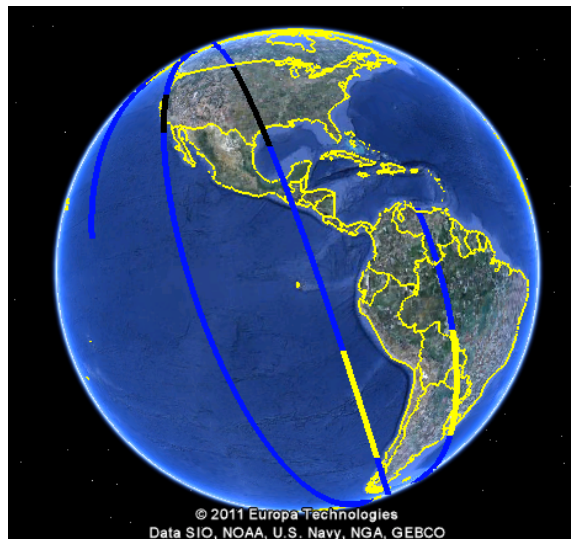


Figura 3- demo_ge_recorrido

Demo_ge_marcador: es la representación de los marcadores. Se han creado dos: Start y Finish que representan el comienzo y el fin del movimiento realizado por el satélite:

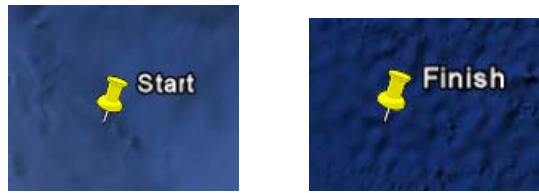


Figura 4- Demo_ge_marcador

Demo_ge_box: es el método que crea el la caja de color rojo, que tiene la función de contener la zona de la captación de las imágenes. El satélite, cerca de esta zona, efectuará tomas que serán guardadas dentro de la misma carpeta:



Figura 5- demo_ge_box

Demo_ge_boxInterno: es una caja interna al precedente. Cuando el satélite entra en la zona roja (box principal) la imagen efectuará un acercamiento hacia la tierra justo para mostrar la zona que captarían las cámaras. El box interno delimita la zona exacta en la cual el satélite hará las fotos:



Figura 6-Demo_ge_boxInterno

Demo_ge_satellite: es la zona donde son representadas las estaciones terrestres. Son pequeñas imágenes de antenas, posicionadas a lo largo del recorrido del satélite:

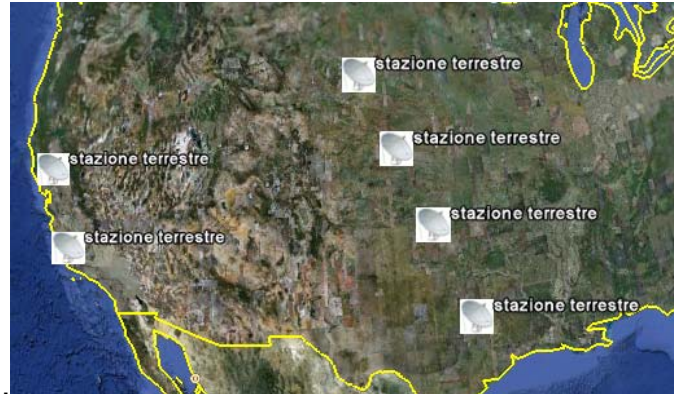


Figura 7- Demo_ge_satélite

Demo_ge_legend: es el método para la creación de la Leyenda. Sirve a resumir las líneas y las geometrías del proyecto, para facilitar su lectura y comprensión para aquellos que lo ven por primera vez:



Figura 8- Demo_ge_legend

Demo_ge_tour: es el método principal. Gracias a esto se ha conseguido reproducir el movimiento del satélite. La dificultad del método ha sido principalmente la resolución de un cálculo matemático que me ha permitido direccionar la cámara hacia un punto determinado.

Google Earth está programado de modo que la cámara indique siempre el norte; así que, una vez cruzado el Polo, la cámara efectúa un giro de 180 grados para diseccionarse hacia el norte de la otra semiesfera.

El cálculo que he realizado permite a la cámara seguir el recorrido sin ningún movimiento. La variable que he tenido que modificar es el *heading*, variable que regula el movimiento de la cámara:

```
a=cosd(latitudine(i))*(longitudine(i)-longitudine(i-1)) / (latitudine(i) - latitudine(i-1));
b= atan(a)*(180/pi);
heading = ['<heading>',num2str(mod(b+180,180)),'</heading>'];
```

La posición de la cámara inicial, ha sido asignada a través de la siguiente línea de código:

```
ge.SetCameraParams(11.1710,-67.7808,2000000,1,10,10,10,10);
```

en orden desde izquierda hacia derecha, los valores representan:

-Latitud: es la distancia angular de un punto del Ecuador, medida a lo largo del meridiano que pasa por aquel punto.

-Longitud: es el ángulo entre el meridiano de un lugar y el meridiano fundamental de Greenwich. Es positiva a Este y negativa a Oeste.

-Altitud: es la distancia vertical de un objeto desde un nivel conocido, llamado nivel cero.

-Modalidad de altitud: representa la manera en la cual se comporta la cámara. Existen dos modalidades:

1. Absolute: indica la altitud del elemento en base al nivel del mar, independientemente de la altura actual del terreno abajo del elemento.

2. Relative: indica la altitud del elemento respecto a la real elevación desde el terreno en un preciso lugar.

-Range: representa la zona de visualización. Más este valor es alto, más grande será la zona enfocada por la cámara.

-Tilt: representa el movimiento vertical de la cámara. Pueden ser utilizados los valores de 0 a 90

-Azimuth: representa el ángulo de la cámara. Puede ser regulada da 0° a 360° para obtener una diferente visión de la tierra.

-Speed: es la velocidad de movimiento del Satélite. Se ha programado de modo que refleje la velocidad real del satélite a lo largo de la órbita preestablecida.

ge_output: es el método citado por todos los demás métodos previamente analizados. Las líneas de código fundamentales son las siguientes:

```

header = ['<?xml version="1.0" encoding="UTF-8"?>',10,...
          '<kml xmlns="http://www.opengis.net/kml/2.2"
          xmlns:gx="http://www.google.com/kml/ext/2.2">',10,...
          '<Document>',10,...
          '<name>',10,name,10,'</name>',10];
footer = [10,'</Document>',10,...'</kml>'];

```

Gracias a estas líneas de código tendremos en todos los ficheros Kml generados la cabecera y el pie del fichero.

En la cabecera se especifica la versión del lenguaje de programación xml que se utiliza.

En todas las funciones se mencionan dos métodos importantes:

- *authoptions*
- *parsepairs*

el primero es un método citado a través de la siguiente línea de código:

```
AuthorizedOptions = authoptions( mfilename );
```

Su fin es aquello de visualizar todas las operaciones posibles y utilizables para aquella función específica.

Al segundo es posible acceder a través de una simple palabra de código llamada:

```
parsepairs
```

este script permite analizar el parámetro y su valor, para comprender si se trata de datos utilizables dentro de la función.

En el fichero principal hay otras tres operaciones importantes que son ejecutadas. La primera es:

```
[lat_sat,long_sat]=demo_ge_funzione;
```

esta línea de código sirve para obtener las coordenadas exactas de la órbita satélital. El cálculo de la órbita se realiza con el método *demo_ge_funzione*, y se entrega al array recién creado. En este método se mencionan variables que conciernen el satélite como la inclinación, la excentricidad, el movimiento, las anomalías, el perigeo, pero sobre todo las dos líneas del TLE de nuestro satélite.

El segundo método importante es *GeServer*. Es el método que regulariza la apertura de Google Earth. Gestiona determinadas excepciones, como en el caso en el cual GoogleEarth no se abra regularmente o no pueda ser abierto porque no existe una conexión a Internet, o cuando la dimensión de la ventana es demasiado pequeña para poder visualizar correctamente el software.

El último método para analizar es *ge.SaveScreenShot* que permite realizar una fotografía de lo que está enfocando la cámara en aquel preciso instante. En el caso específico he utilizado la siguiente línea de código:

```
ge.SaveScreenShot('C:\Users\Antonio\Documents\MATLAB\Tes\foto.kmz',50);
```

3. Conclusiones

En la figura 9 es posible visualizar el resultado completo de la aplicación. Todas las funciones y los métodos mencionados a partir del expediente *Antonio*, generan unos ficheros *kml* que permiten ver la animación. El resultado se puede ver en la figura 9:

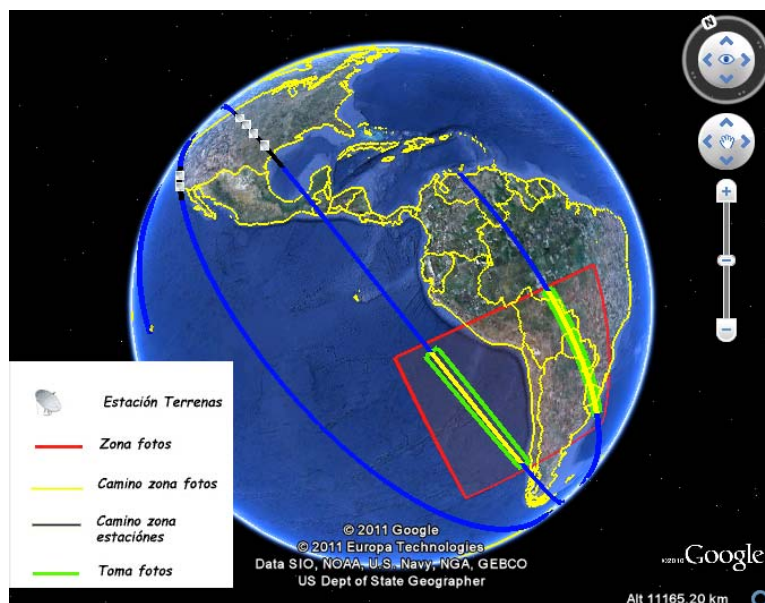


Figura 9-Panorámica sobre la aplicación

Por supuesto, lo único que no puede ser visualizado es el 'tour'. Sin embargo hay una representación gráfica a través de una serie de imágenes con las cuales es posible intuir el movimiento de la cámara a lo largo del recorrido de la órbita:



inicio: es el inicio del 'tour'. La cámara se pone en posición, en dirección y ángulo



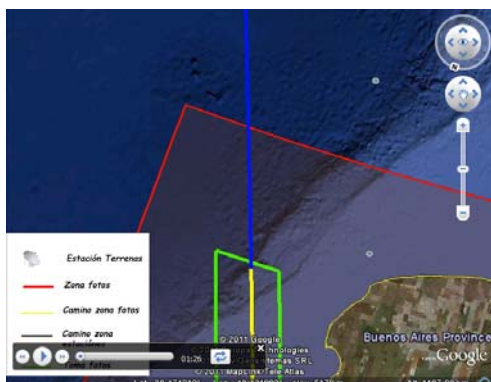
El 'tour' continua atravesando los países de Sur América



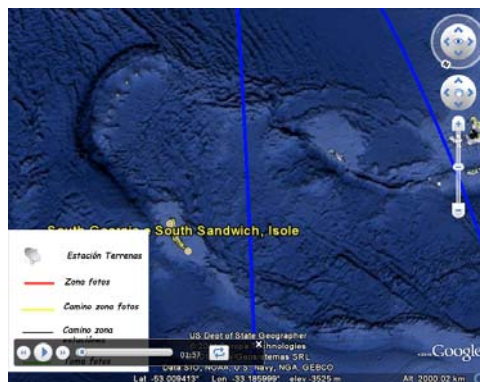
vamos a visualizar el acercamiento hacia la zona de adquisición de la imagen



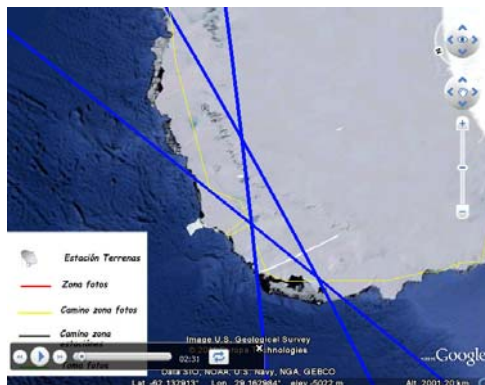
zona de adquisición de la imagen



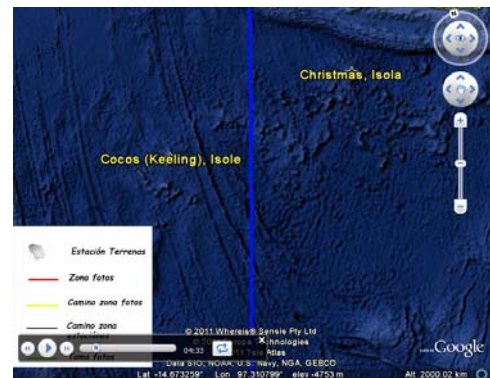
alejamiento desde la zona de adquisición de las imágenes



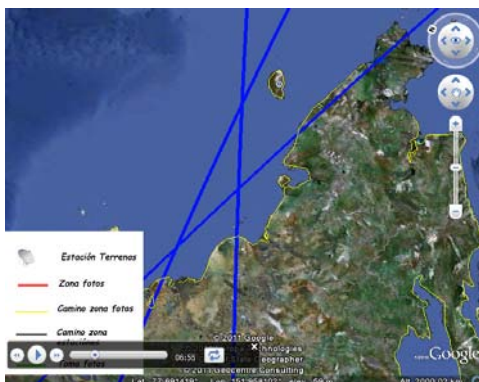
paso por las islas: Georgia del Sur y South Sandwich



paso por el polo Sur



paso por las islas de Cocos



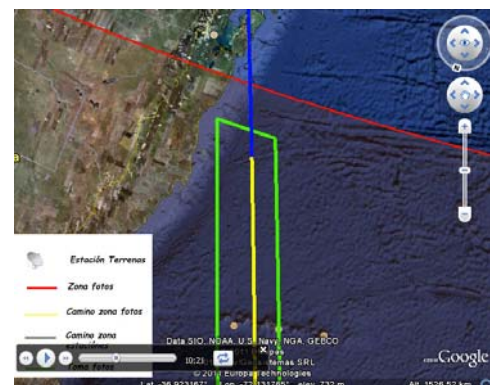
el satélite atraviesa por primera vez el polo Norte



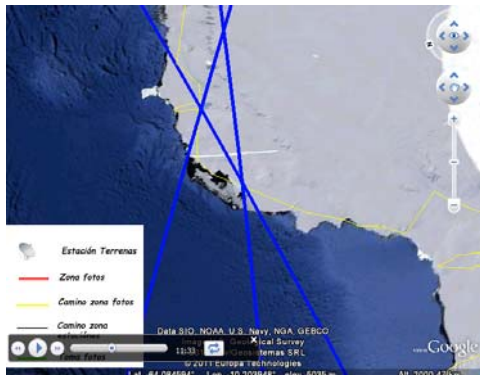
Estados Unidos: zona de posición de las estaciones terrestres



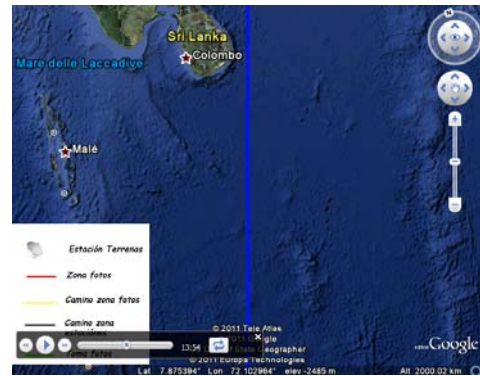
acercamiento a la zona de adquisición de las imágenes



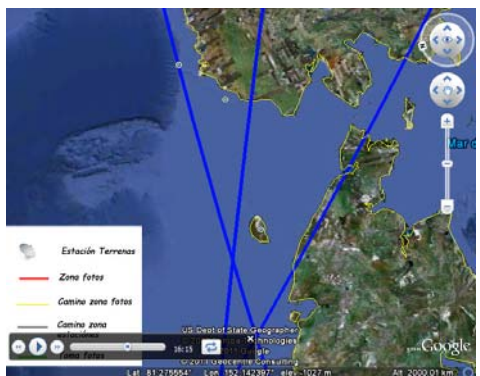
alejamiento desde la zona de adquisición de las imágenes



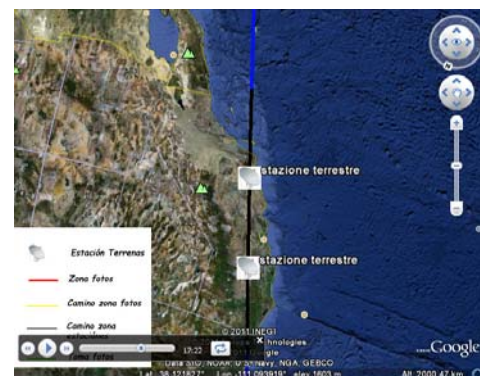
segundo paso del polo Sur



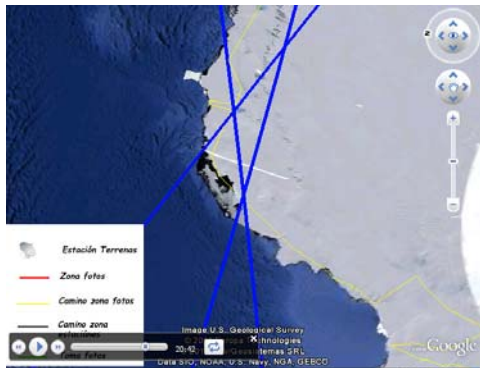
inicio del paso por Asia



segundo paso por polo Norte



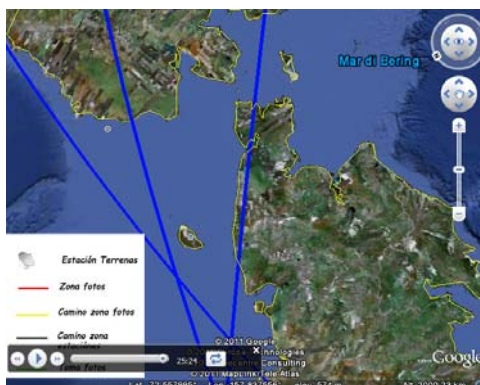
segundo paso por las estaciones terrestres



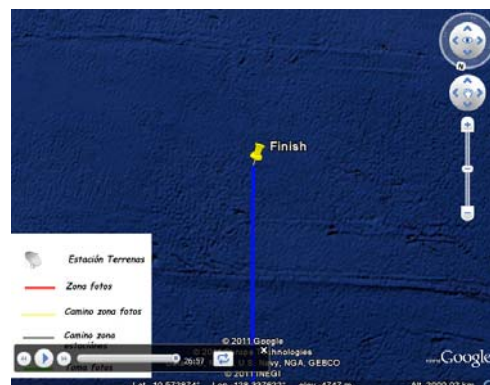
tercer paso por el polo Sur



paso por Kazakhstan



último paso del polo Norte



fin: se acaba nuestro 'tour'

El desarrollo de este proyecto, durante estos meses, ha sido algo muy útil desde el punto de vista profesional, laboral y personal.

Además trabajar a este proyecto me ha permitido estudiar y aprender a utilizar de la mejor manera posible software tan importante como Matlab y Google Earth.

Matlab es un software que hemos utilizado poco durante los primeros tres años de la Universidad. Está enfocado al cálculo numérico y al análisis estadístico pero permite visualizar e implementar funciones, datos y algoritmos.

He estudiado el lenguaje de programación creado por la Mathworks. He aprendido como utilizar los métodos, como crear variables, array, funciones, aunque la parte mas difícil ha sido conectar Matlab con otro software importante como Google Earth. De este software he tenido que estudiar las API. Comprender como están realizadas las funciones, entender cómo y qué parámetros utilizan las funciones, y sobre todo comprender como poder crear un fichero Google Earth desde Matlab.

Todo esto ha contribuido a la realización de mi proyecto, pero la cosa más útil, ha sido entender la importancia que tiene organizar el trabajo, el tiempo, utilizar Internet, leer libros, estudiar material y luego aplicar lo que se ha aprendido para así poder desarrollar el proyecto.

Creo haber conseguido cumplir y alcanzar los objetivos que me había planteado al empezar el proyecto, además de haber obtenido unos buenos resultados finales.

3.1 Futuras expansiones

A partir del proyecto por mi desarrollado se pueden preveer y plantear expansiones futuras.

- Se podría crear un menú en Matlab que permitiera controlar la cámara, bloquear y reiniciar el movimiento utilizando un control

directamente de Matlab, en vez de utilizar el control de Google Earth.

- Se podría crear una doble cámara, con el fin de elegir o alternar y tener así una visualización diferente del recorrido orbital a la vez que se procede con la misma.
- También se podría crear un pequeño menú que permitiría elegir entre dos o tres diferentes órbitas y visualizarlas una después de otra.

Son todas aplicaciones y mejoras que podrían ser utilizadas en proyectos futuros.

4. Bibliografía

- 1: G. Naldi, L. Pareschi, "Matlab: Concetti e progetti" .
- 2: Documentazione di Matlab: www.mathworks.com .
- 3: Documentazione Api Google Earth: www.earth.google.com/comapi/ .
- 4: C. Azqueta, "Software de tierra para la planificacion orbital de la actitud de satelites".
- 5: K. Howard Goldberg, E. Castro, "XML" .